

How to use the 2068 User Cartridge Board

On power up, the TS2068 looks at certain locations (hereafter called "AROS overhead bytes") in the ROS (Rom Oriented Software) bank of its memory space to see if a cartridge is installed in its cartridge port. If it finds the required data, it uses the information to set itself up for that particular cartridge. The data contained here, from 8000H to 8007H for AROS (Application Rom Oriented Software), tells the system the 8K memory "Chunks" used by the cart. within the ROS (cartridge) bank, the language the software is in (Basic & MC or MC only), etc. It should be noted here that the system also checks the bottom of the ROS bank for LROS firmware too, (Language Rom Oriented Software) but this particular application is NOT intended for the typical user of this "User Cartridge Board", but is for additional languages on the 2068. We will be concerned only with AROS in our applications.

Because the AROS overhead bytes occupy 8000-8007H, the first byte available for our programs is 8008H. To be efficient, we will start ALL of our Basic programs at 8008H, and I recommend starting a MC only program here also. In this note, however, we will cover only Basic language programs-I will leave the MC only use of the board up to you!

The following are the AROS overhead bytes' locations, along with the particular locations' function in the system and the recommended values for a typical Basic program. These are the actual values we will be using for our purposes. MOST Basic programs will run directly out of our cartridge AS IS, but a few may require slight modifications for functions not supported by the 2068 cartridge system. The only two functions I have so far found that are NOT supported on a cartridge that ARE supported in a ram based program are: 1) User defined functions (DF FN & FN) & 2) A non-executing FOR/NEXT loop such as FOR N=2 TO 0, which will return with Report "I" from AROS.

ADDRESS/FUNCTION	MEANING	OUR VALUE
8000=Language type	1=Basic 2=MC only	01H Basic
8001=Cartridge type	1=LROS 2=AROS	02H AROS
8002/8003=Prog start add	LSB/MSB of starting address	0880H Prog begins @8008H
8004=8k Chunk spec	Bits 0-7=8k blks 0-7 used if 0	0FH 8000-FFFFH in use
8005=Autostart spec	0=Not autostart or 1=autostart	01H Autostart
8006/8007=????	Bytes in ram bank reserved	0000H All of ram avail.

Thus the first 8 bytes of our eprom starting at location 8000H in the cartridge board should be: 01,02,08,80,0F,01,00,00 and our Basic program should start at location 8008H. It should be noted here that ALL of the normal Home ram mapped on top of our cartridge from 8000-FFFFH can and will be used by the system for data storage of program variables and arrays. Thus our DATA SIZE is NOT reduced by our Basic program length in the ROS bank on eprom.

There is one more detail concerning the cartridge system I would like to bring up that I have discovered. If you BREAK or STOP a Basic program on cartridge, you do NOT see the Basic listing on the cartridge, nor can you enter Basic lines into ram, EDIT, SAVE/LOAD from immediate mode, etc. without problems. If you enter RUN, the program in the cartridge will be run...same thing happens with GOTO XXXX.

I have found that bit 7 of system variable 23750 is the software "switch" for the cartridge system on the 2068. If you POKE 23750,0 then you will be able to enter Basic lines into ram, edit them, delete them, run THEM, etc.. If you POKE 23750,128 then you will NOT be able to do these things and a RUN or GOTO will apply to the program in the cartridge board. Thus 23750 appears to be the key back & forth between the cartridge program and a ram based program. This, again, is only something I have discovered & NOT God's law, so experiment with this after you have a cartridge up & running. Write me if you find out any other interesting things on the cartridge system of the TS2068. Your discoveries may help another TS2068 enthusiast somewhere!!!

To program your eprom you will need either the Oliger TS1000 2764 Programmer and one of the Oliger TS1000 Expansion Boards or the Oliger 2068 Eprom Programmer and the Oliger 2068 Expansion Board. The smaller TS1000 expansion board, if used, should be inserted into the 2068 with its key in line with the slot on the rear edge traces of the 2068, making it contact the center 23 pins of the 2068 bus. The TS1000 2764 programmer can also be used with the 2068 Expansion Board, if desired, by again lining up its slot with the key in any of the 2068 expansions connectors and plugging it in. Plug the Oliger Expansion Board into your 2068 computer and one of these programmers, with fresh eprom installed in its programming socket, into the expansion board.

We will cover the programming of 2764s only in THIS documentation, using either the Oliger TS1000 2764 Programmer or the Oliger 2068 Programmer with its select switch set to "64". Both programmers require exactly the same software, when the 2068 board is set to "64". Refer to your Oliger 2068 Programmer user manual for details of programming 27128 eproms for use with this board.

To put your program into the eprom after you have the programmer attached & ready, simply LOAD in your Basic program. After LOADING, I suggest removing all REM statements & eliminating all user defined functions (DEF FN & FN) After your program is edited as desired for the cart., key in <CLEAR><ENTER> then <PRINT PEEK 23627 + PEEK 23628 * 256><ENTER>. Write down 1 less than the number printed on the screen. If this number is greater than 34,893 then your program is longer than the 8K & will require two 2764 eproms to hold it. If it is just a little more than 34893 you may be able to DELETE some of the lines or compact the program by using more statements per Basic line. You can try DELETing and editing to get the number no greater than 34893, but if unsuccessful must use two 2764 eproms to hold the program. The number MUST be no more than 43095 or it will not even fit on two 2764s! You are limited to a Basic LISTING no greater than 16K in length unless using 27128s with the 2068 Programmer.

What follows now is the actual Basic program to put your program, less this addition, onto a 2764 eprom. Use the number you wrote down for XXXX in line 9994 if not greater than 34893. If the number WAS greater than 34893 then use 34893 for XXXX this time-you will have to program a second 2764 for the other half later, at which time you will then use the number you have written down for XXXX and 34894 to replace 26710 in line 9994.

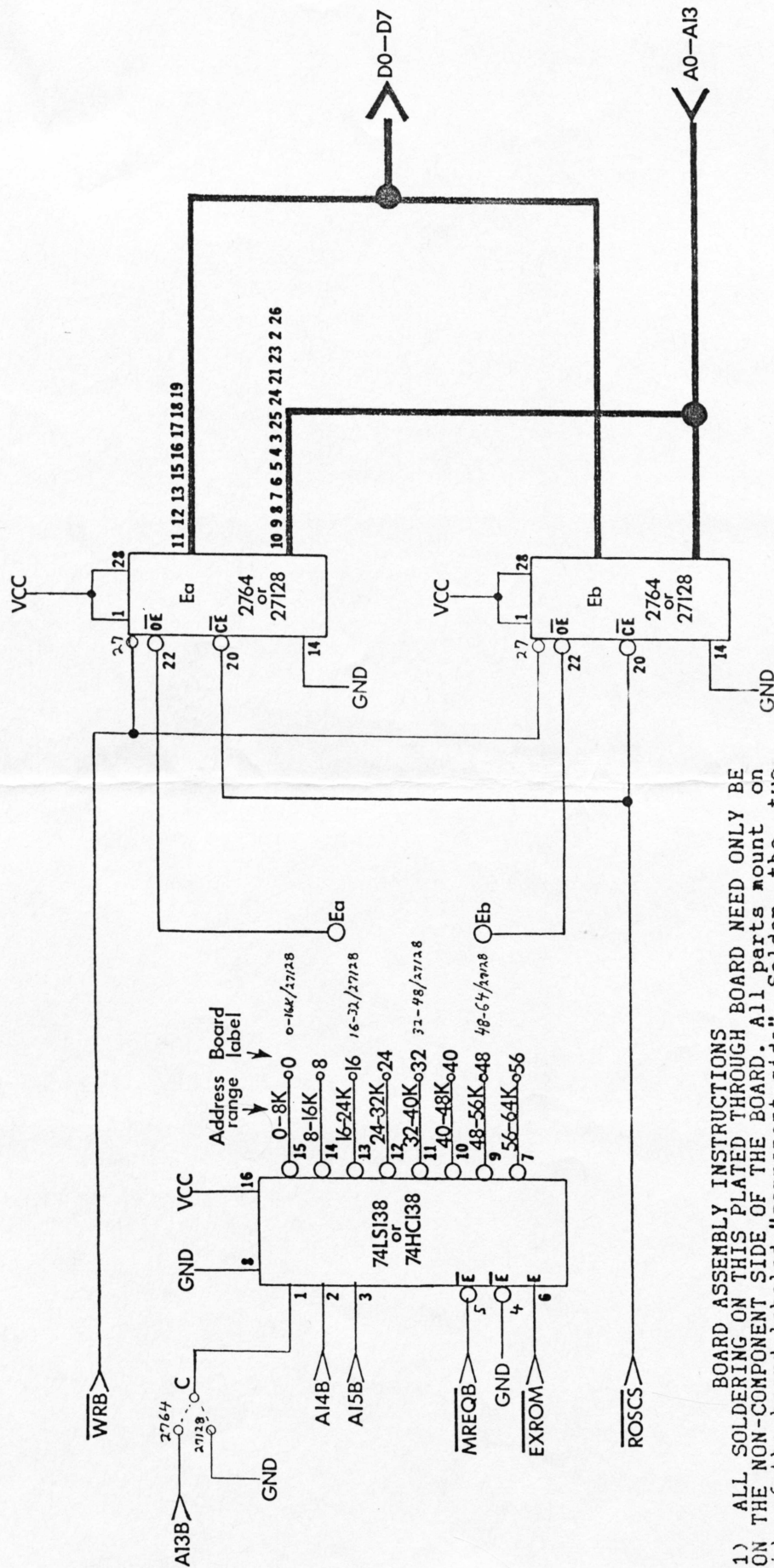
```
9990 LET X=8192
9991 RESTORE 9993 : FOR N=1 TO 8 : READ Z : POKE X,Z
9992 PAUSE 4 : LET X=X+1 : NEXT N
9993 DATA 1,2,8,128,15,1,0,0
9994 FOR N=26710 TO XXXX (No. written down)
9995 POKE X, PEEK N : PRINT AT 10,11,N
9996 LET X=X+1 : PAUSE 3 : NEXT N
9997 BEEP 1,1
```

Use <GOTO 9990><ENTER> to start the eprom programming after the Vpp supply is switched to 21 Volts.

After the BEEP has sounded, signalling that our eprom is programmed, lower and remove Vpp, downpower the computer, & install this eprom in the Ea socket of the 2068 User Cartridge board which should be mapped 32-48K. (VIA jumpering the "32" pad of the decoder chip to the doughnut labeled "Ea") If another eprom is required because the program was greater than 8K in length, then program it now. (You do not use lines 9991-9993 when programming the second eprom) When the entire program is on eprom (Eb should be mapped 40-48K IE: Its "Eb" doughnut should be connected to the 138s "40" pad) & installed on the cartridge board, we are now done and ready to try out our new User Cartridge! With the computer off, insert the User Cartridge board into the 2068's cartridge socket, then power up your computer. The program you put on the eprom(s) should come up and running. If it does HOOORAAAY!!!! If it doesn't (BOOO!!) then an error has been made somewhere & you will have to erase the eprom(s) and try again. You may consider trying a very short "test" program to get the feel of the procedure and gain some familiarity with it. GOOD LUCK and HAVE FUN!!!

TS2068 USER CARTRIDGE BOARD

REV. 'A'



BOARD ASSEMBLY INSTRUCTIONS

STEP 1) ALL SOLDERING ON THIS PLATED THROUGH BOARD NEED ONLY BE DONE ON THE NON-COMPONENT SIDE OF THE BOARD. All parts mount on the side of the board labeled "component side". Solder the two 28 pin IC sockets onto the board. Use extra care soldering the narrow pads surrounded by board traces.

STEP 2) Solder the 74LS138/74HC138 IC in place with pin one oriented towards the "1" label etched onto the board. DO NOT use an IC socket for this chip or the board will not clear the TS2068's case upon insertion into your computer.

STEP 3) Use acetone and a soft cloth or a commercial flux remover to clean all traces of flux from the board. CAREFULLY inspect all of your soldering, hunting for poorly soldered joints or solder blob shorts to adjacent traces. Touch up anything found looking even remotely suspicious.

STEP 4) Use a short piece of stripped wire/wrap for 2764s or from the pad labeled "C" to the pad labeled "64" for combinations "128" for 27128s or 2764/27128 combinations. For combinations the 27128 must go in the "Ea" socket. Use short pieces of wire/wrap wire to jumper from the 138 pad labeled "32" to the pad labeled "Ea". Jumper from the 138 pad labeled "40" to pad "Eb" (for 2764s) or from "48" to "Eb" (for 27128s or 2764/27128 combinations). All of these jumpers are installed on the solder side of the board. Your Olliger User Cartridge is now ready!

PARTS LIST:

1pc...74LS138 or 74HC138 IC
2pcs...28 pin low profile IC socket
1pc...2 inches 30AWG wire/wrap wire
2pcs...2764 or 27128 Eprom, 250ns or faster (user supplied)

The John Olliger Co.
11601 Whidbey Dr.
Cumberland, IN 46229
All rights reserved